

Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application.

Listing of Claims:

1. (Currently Amended) A method for converting display source code of a ~~non-web~~ enabled legacy application having mixed business and presentation logic on a server to a network interactive web-browser page, the method comprising the steps of:
 - (a) resolving the display source code of the ~~non-web-enabled~~ legacy application into a plurality of record formats;
 - (b) for each of the plurality of record formats, resolving a plurality of references within the record format to database files;
 - (c) determining the ~~a hierarchy and relationships~~ relationship of the plurality of references within the database files;
 - (d) parsing the plurality of references within the database files to a web-language file using nested tags to capture the hierarchy and relationship of the plurality of references to create one or more network user interface pages; and
 - (e) converting the one or more network user interface pages to an object-oriented platform-independent network language by including:
 - (i) creating dynamic components for input, output, and feedback references of each record format, the dynamic components for populating the one or more network user interface pages; and
 - (ii) creating a static component for unchanging references of each record format, the static component representing the one or more network user interface pages.

2. (Original) The method of claim 1, wherein the network interactive web-browser page is displayed on the Internet.

3. (Original) The method of claim 1, wherein the network interactive web-browser page is displayed on a network selected from the group consisting of: an internal network, an Intranet, a LAN, a WAN, an internal bus, a wireless network.

4. (Original) The method of claim 1, wherein the web-language file is an XML language file.

5. (Original) The method of claim 4, wherein the XML language file is an HTML file.

6. (Original) The method of claim 4, wherein the XML language file is a WML file.

7. (Original) The method of claim 1, wherein the static component further comprises a JavaServer Page.

8. (Original) The method of claim 1, wherein the dynamic components further comprise JavaBeans.

9. (Currently Amended) The method of claim 2, wherein the one or more network user interface pages are stored on the server.

10. (Currently Amended) A computer readable medium containing program instructions for creating web interfaces of a ~~non-web-enabled~~ legacy application having mixed business and presentation logic stored on a computer, the program instructions for:

- (a) parsing display file data description source of the ~~non-web-enabled~~ legacy application to render the source into a plurality of network user interface pages, each network user interface page to display a record format of the ~~non-web-enabled~~ legacy application;
- (b) converting the each network user interface page, wherein at least one data object maintains the ~~non-web-enabled~~ application's input data, output data, and feedback data of the legacy application on a client, and at least one web-browser page maintains the ~~non-web-enabled~~ application's static content of the legacy application;
- (c) dynamically updating the web-browser page with the ~~non-web-enabled~~ application's input data, output data, and feedback data of the legacy application via a servlet instance; and
- (d) displaying the dynamically updated web-browser page on a client via a network.

11. (Currently Amended) A computer readable medium containing program instructions for use in a computer network, the computer readable medium containing program instructions for:

- (a) providing a plurality of network user interface pages of ~~format-records~~ record formats of display source code of a ~~non-web-enabled~~ legacy application having mixed business and presentation logic, the network user interface pages to receive data from the ~~non-web-enabled~~ legacy application and in response thereto; and
- (b) converting the network user interface pages to web-browser pages, a static portion of which displays the static portion of the ~~format-record~~ record formats and a dynamic portion of

which interacts with the web-browser page to display input data, output data, and feedback data required by ~~from/of~~ by and from the ~~non-web-enabled~~ legacy application.

12. (Currently Amended) A computer system for executing an application, comprising:

- a central processing unit;
- a main memory connected to the central processing unit with a communication bus;
- a data storage unit connected to a data storage interface which is connected to the communication bus;
- at least one input/output device connected to the communication bus and connected to a network interface to an external computer network,
- a ~~non-web-enabled~~ legacy application having mixed business and presentation logic stored in the main memory and capable of executing on the central processing unit; and
- a plurality of intermediate network user interface pages, each of which correspond to a record format of the ~~non-web-enabled~~ legacy application;
- wherein as the ~~non-web-enabled~~ legacy application executes, application logic may use either a ~~non-web-enabled~~ legacy application display of the record format or the plurality of intermediate network user interfaces pages for communication of the ~~non-web-enabled~~ legacy application to a user over the external computer network.

13. (Currently Amended) A computer server for converting the display source of a ~~non-web-enabled~~ legacy application having mixed business and presentation logic stored and executing on a computer, comprising:

- a central processing unit;

a parser to parse the display source of the legacy application into a plurality of record formats, each of the record formats being unique to each input/output screen definition of the ~~non-web-enabled~~ legacy application;

a generator of web-language user interface files having nested tags of each of the record formats; and

a converter ~~of operable to convert the~~ web-language user interface files to one or more web pages, the converter further comprising:

an object creator to create dynamic components for the dynamic portions of the ~~display source record formats, the dynamic components for populating the one or more~~ web pages; and

a static component for display of the unchanging aspects of the record formats, the static component representing the one or more web pages.

14. (Currently Amended) The method of claim 1, wherein each step of the steps ~~(a)-(e)~~ method occurs at development time during which a user is preparing a new user interface for the ~~non-web-enabled~~ legacy application.

15. (Currently Amended) The method of claim 14, wherein ~~step (e)~~ converting the one or more network user interface pages is ~~further operable to be~~ performed offline without any remote connection to the server upon which the ~~non-web-enabled~~ legacy application resides.

16. (Currently Amended) The computer readable medium of claim 10, wherein:
the program instructions for parsing the display file data description source and
converting the network user interface page are each executed during development time of the

web-browser page; and

the program instructions for dynamically updating the web-browser page and displaying the

web-browser page are each executed during runtime. ~~of the non-web-enabled application.~~